Universidad
Carlos III de Madrid

# PROGRAMMING FOR BUSINESS

Name: Nicole Vásquez (100381657)
Date: May, 18th 2020
Professor: Fuensanta Medina

**EXERCISE SHEET 1**

**EXERCISE 1**

**<u>While loop</u>**

```
Calculatedata<-function() {

  grades=matrix(c(8,9,3,8,2,6),nrow=3,ncol=2,byrow=T,)

  rownames(grades)=c("Gema","Rosa","Lucia")

  colnames(grades)=c("Programming","Math")

  print(grades)


i = 1

  while (i <= ncol(grades)){

    sub =colnames(grades)[i]

    print(paste("The max and min grade of",sub, "are", max(grades[,i]),"and", min(grades[,i])))

    i = i + 1
}


print(paste("The max and min grade of both subjects are: ", max(grades), "and", min(grades)))


 print(nrow(subset(grades, grades[,1] >=5 & grades[,2] >= 5)))

  ask=as.numeric(readline(prompt="Please, enter a grade: "))

  if(ask>=0 & ask<=10){

    print(length(which(grades %in% ask)))

  }

}

Calculatedata()
```

## Repeat loop

```
Calculatedata<-function() {
  grades=matrix(c(8,9,3,8,2,6),nrow=3,ncol=2,byrow=T)
  rownames(grades)=c("Gema","Rosa","Lucia")
  colnames(grades)=c("Programming","Math")
  print(grades)

i = 1
 repeat{
    sub =colnames(grades)[i]
    print(paste("The max and min grade of",sub, "are", max(grades[,i]),"and", min(grades[,i])))
    i = i + 1
    if (i>ncol(grades)){
      break
  }
  }
  print(paste("The max and min grade of both subjects are: ", max(grades), "and", min(grades)))

  print(nrow(subset(grades, grades[,1] >=5 & grades[,2] >= 5)))

  ask=as.numeric(readline(prompt="Please, enter a grade: "))
  if(ask>=0 & ask<=10){
    print(length(which(grades %in% ask)))
  }
 else{
    print("Not valid grade")
  }
}
Calculatedata()
```

## For loop

```
Calculatedata<-function()
{
 grades=matrix(c(8,9,3,8,2,6),nrow=3,ncol=2,byrow=T,)
 rownames(grades)=c("Gema","Rosa","Lucia")
 colnames(grades)=c("Programming","Math")
 print(grades)

 for(i in (ncol(grades))){
  sub =colnames(grades)[i]
  print(paste("The max and min grade of",sub, "are", max(grades[,i]),"and", min(grades[,i])))
  i = i + 1
 }
 print(paste("The max and min grade of both subjects are: ", max(grades), "and", min(grades)))
 print(nrow(subset(grades, grades[,1] >=5 & grades[,2] >= 5)))

  ask=as.numeric(readline(prompt="Please, enter a grade: "))
 if(ask>=0 & ask<=10){
  print(length(which(grades %in% ask)))
 }
}
Calculatedata()
```

**EXERCISE 2**

```
StudentData<-function()
{data=matrix(c(2.5,2,0.3,2.5,2.1,1.2,2.5,2,1.6,0.9,2.4,2.8),ncol=2)
rownames(data)=c("Ana","Pepe","Nacho","Bea","Gema","Alba")
colnames(data)=c("Partial 1","Partial 2")
data=cbind(data,c(rowSums(data)))
colnames(data)=c("Partial 1","Partial 2","Grades")
Submit=c("YES","YES","NO","YES","YES","YES")
data=cbind(data,Submit)
data=rbind(data,c(2.3,2.1,4.4,"YES"))
rownames(data)=c("Ana","Pepe","Nacho","Bea","Gema","Alba","Javier")
i=1
FG=c()
while(i<=nrow(data)){
  D=as.numeric(data[i,3])
  if(data[i,4]=="YES"){
    D=D+1
    FG=c(FG,D)
    i=i+1
  }
  else{
    if(data[i,4]=="NO"){
      D=D+0
      FG=c(FG,D)
      i=i+1
    }
  }
}
data=cbind(data,FG)
print(data)

pass=0

nopass=0

for(i in 1:nrow(data)){

  if(data[i,3]>=5){

    pass=pass+1

  }

  else{

    nopass=nopass+1

  }

}

cat("Number of students who passed: ",pass,"\n","Number of students who didn'd pass: ",nopass)

}

StudentData()
```

## EXERCISE SHEET 2

### EXERCISE 1

**Using the sort ().**

```
ex1.1<- function()
{A=c(4,1,8,2,9,6,3)
num=as.numeric(readline(paste("Please, enter a number between",min(A),"and", max(A),": ")))
if(num<min(A) | num>max(A)){
  print("The number is not valid")
}
else
  if(num %in% A){
    print(sort(A))
  }
else
{
  A=c(A,num)
  print(sort(A))
}
}
ex1.1()
```

**Without using the sort ().**

```
ex1.2<- function()
{A=c(4,1,8,2,9,6,3)
num=as.numeric(readline(paste("Please, enter a number between",min(A),"and", max(A),": ")))
if(num<min(A) | num>max(A)){
  print("The number is not valid")
}
else if(num %in% A){
  for(i in 1:(length(A)-1)){
   for(j in (i+1):length(A)){
    if(A[i]>A[j]){
     A[c(i,j)] <- A[c(j,i)]
    }
   }
  }
  print(A)
}
else{
  A=c(A,num)
  if(num %in% A){
   for(i in 1:(length(A)-1)){
    for(j in (i+1):length(A)){
     if(A[i]>A[j]){
      A[c(i,j)] <- A[c(j,i)]
     }
    }
   }
  }
  print(A)
}
}
ex1.2()
```

## EXERCISE 2

```
Calculatedata<-function()
  {grades<-array(c(8,3,2,9,8,6),c(3,2,1))
dimnames(grades)=list(c("Gema","Rosa","Lucia"),c("Programming","Math"))
print(grades)


for(i in 1:ncol(grades)){
  sub =colnames(grades)[i]
  print(paste("The max and min grade of",sub, "are", max(grades[,i,]),"and", min(grades[,i,])))
  i = i + 1
}
print(paste("The max and min grade of both subjects are: ", max(grades), "and", min(grades)))


print(length(subset(grades, grades[,1,] >=5 & grades[,2,] >= 5)) / ncol(grades))


ask=as.numeric(readline(prompt="Please, enter a grade: "))
if(ask>=0 & ask<=10){
  print(length(which(grades %in% ask)))
}
}
Calculatedata()
```

**EXERCISE 3**

```
Hit<-function()

{A=c(2,3,6,1,7,1)

v = array("*",length(A))

print(A)

print(v)

while ("*" %in% v){

num=as.numeric(readline(prompt="Please, enter a number between 1 and 10: "))

if(num<1 | num>10){

  print("The number is not valid.")

}

else

  if(num %in% A){

    v[which(A %in% num)] <- as.character(num)

    A[which(A %in% num)] <- "*"

    print(A)

    print(v)

  }

else{

  print(A)

  print(v)

}

}

}

Hit()
```

**EXERCISE 4**

**with while**

```
 CalculateHeightandWeights<-function()

{data=array(c(1.70,1.65,1.80,70,70,76),c(3,2,1))

dimnames(data)=list(c("Pepe","John","Antonio"),c("Height","Weight"))

print(data)
i=1
while(i<ncol(data))
print(paste("The minimum height and weight are: ", max(data), "and", min(data)))

  i=i+1

}

h=0
D=apply(data,2,mean)
i=1
while(i<=nrow(data)){

  if(data[i,1,]>D[1])

    h=h+1

    i=i+1

}

print(D)
print(h)

j=1
w=0

D=apply(data,2,mean)

while(j<=nrow(data)){

  if(data[j,2,]>D[2])

    w=w+1

  j=j+1

}

print(w)

}

CalculateHeightandWeights()
```

**With for**

```
CalculateHeightandWeights<-function()
{data=array(c(1.70,1.65,1.80,70,70,76),c(3,2,1))
dimnames(data)=list(c("Pepe","John","Antonio"),c("Height","Weight"))
print(data)
for(i in 1:(ncol(data)-1)){
  print(paste("The minimum height and weight are: ", max(data), "and", min(data)))
}
h=0
D=apply(data,2,mean)
for(i in 1:nrow(data)){
  if(data[i,1,]>D[1])
    h=h+1
}
print(D)
print(h)
j=1
w=0
D=apply(data,2,mean)
for(j in 1:nrow(data)){
  if(data[j,2,]>D[2])
    w=w+1
}
print(w)
}
CalculateHeightandWeights()
```

**With repeat**

```
CalculateHeightandWeights<-function()
{data=array(c(1.70,1.65,1.80,70,70,76),c(3,2,1))
dimnames(data)=list(c("Pepe","John","Antonio"),c("Height","Weight"))
print(data)
i=1
repeat{
  print(paste("The minimum height and weight are: ", max(data), "and", min(data)))
  i=i+1
  if(i>=ncol(data))
    break
}
h=0
D=apply(data,2,mean)
i=1
repeat{
  if(data[i,1,]>D[1])
    h=h+1
  i=i+1
  if(i>nrow(data))
    break
}
print(D)
print(h)
j=1
w=0
D=apply(data,2,mean)
repeat{
  if(data[j,2,]>D[2])
    w=w+1
  j=j+1
  if(j>nrow(data))
    break
 }
print(w)
}
CalculateHeightandWeights()
```

**EXERCISES SHEET 3**

**EXERCISE 1**

```
ex1<-function(x,y){
  if (is.vector(y) & length(y) >= 2 & is.numeric(x)){
for(i in 1:length(y)){
  if(y[i]==x){
    print("The number is in the vector")
break
    }
}
  }
  else{
    print("y it’s not a vector or x it’s  not a number")
  }
}
ex1()
```

**EXERCISE 2**

```
ex2<-function(x,y){

 if (is.array(y) &  is.numeric(x)){

  for(i in 1:length(y)){

   if(y[i]==x){

    print("The number is in the array")
 break
}
}
}
else{

  print("y it’s not an array or it’s not a number")

 }

}
ex2()
```

**EXERCISE 3**

```
ex3<-function(x,y){

 if (is.array(y) &  is.numeric(x)){

  for(i in 1:length(y)){

   if(y[i]==x){

    print("The number is in the array")

    D=length(which(y %in% x))
print(D)
    break

   }

  }

 }
 else{
 print("y it’s not an array or it’s not a number ")

 }
}

ex3()
```

## EXERCISES SHEET 4

### EXERCISE 1

```
area<-function()
 {side=as.numeric(readline(prompt="Please, enter the length of the sides of the square: "))
 A=side
 A=A^2
 cat("The area of the square is: ", A)
}
area()
```

### EXERCISE 2

```
mi.factorial <- function(n){
 factorial <- 1
 for (i in 1:n){
  factorial <- factorial * i
 }
 return(factorial)
}
mi.factorial(n)
```

### EXERCISE 3

```
int<-function()
 {ask=as.numeric(readline(prompt="Please, enter a integer between 0 and 9:"))
 num=(sample(0:9,1))
 if(num==ask){
  print("You are correct.")}
else
 cat("The number is not correct, the correct number is: ",num)

}
int()
```

**EXERCISE 4**

```
int2<-function()
{num=(sample(0:9,1))
i=1
while(i<2){
 ask=as.numeric(readline(prompt="Please, enter a integer between 0 and 9:"))
 if(num != ask){
  print("The number is not correct. Please, enter another number.")}
 else
  if(num==ask){
   print(paste("You are correct, the number is ",num))
   i=i+1
  }
}
}
int2()
```

**EXERCISE 5**

```
mat<-function()
{x=matrix(1:10,nrow=2)
print(x)
  num1=as.numeric(readline(prompt="Please, enter a number between 1 and 10: "))
 if(num1<1 | num1>10){
  print("The number is not correct")
 }
 else
  if(num1 %in% x){
   x[which(x%in%num1)]=-1
 print(x)}
  num2=as.numeric(readline(prompt="Please, enter a number between 1 and 10: "))
  if(num2<1 | num2>10){
    print("The number is not correct")
  }
  else
    if(num2 %in% x){
      x[which(x%in%num2)]=-1


      print(x)
    }
}
mat()
```

**EXERCISE 6**

```
mat2<-function()
{x=matrix(c(1,2,3,4),ncol=2)
cat("Matrix 1: ","\n")
      print(x)
      cat("Matrix 2: ","\n")
y=t(x)
print(y)
cat("Sum of both matrices:","\n")
z=x+y
print(z)
}
mat2()
```

**EXERCISE 7**

```
mat3<-function()
{A=matrix(c(1,3,2,4),nrow=2,byrow=T)
print(A)
B=matrix(c(1,3,2,4),nrow=2,byrow=T)
print(B)
D=A[which(A %in% B)]
for( i in 1:length(D)){
  A[which(A %in% D[i])] <- -1
  B[which(B %in% D[i])]<- -1
}
print(A)
print(B)
}
mat3()
```

**EXERCISE 8**

```
ex8<-function(x,y){
  if (is.matrix(y) &  is.numeric(x)){
      for(i in 1:length(y)){
        if(y[i]==x){
          print("The number is in the matrix")
          D=length(which(y %in% x))
          print(D)
          break
        }
      }
    }
```

```
   else{
     print("y it's not a matrix or x is not a number")
   }
 }
ex8()
```

### EXERCISE 9

```
sumMatrix<-function()
{  A=matrix(c(sample(1:10)),nrow=2)
print(A)
sumrows=(rowSums(A))
sumcolumns=(colSums(A))
print(sumrows)
print(sumcolumns)
}
sumMatrix()
```

### EXERCISE 10

```
contabilizarNumMatriz<-function()
{A=matrix(c(1,3,3,4),ncol=2)
print(A)
linea <- NULL
for( i in 1:length(A)){
  N=length(which(A%in% A[i]))
  B=A[i]
  linea1 <- c(B, N)
  linea <- rbind(linea,linea1)
}
print(unique(linea))
}
contabilizarNumMatriz()
```

**EXERCISE 11**

```
vect<-function()
 {A=(sample(1:10,5))
 print(A)
 B=c(4,1,7,3,2)
 print(B)
 matching=c()
 nonmatching=c()
 for(i in 1:length(A)){
  for(j in 1:length(B)){
    if(A[i]==B[j]){
    matching=c(matching,A[i])
    }
  }
 }

 print(sort(matching))
 nonmatching = A[-which(A %in% matching)]
 print(sort(nonmatching))
 }
vect()
```

**EXERCISE 12**

```
ex12<-function()
{A=(sample(1:10,4))
B=c(5,2,3,4)
even=c()
odd=c()
print(A)
print(B)
for(i in 1:length(A)){
if((A[i]%%2)==0){
  even=c(even,A[i])}
else{
  if((A[i]%%2)==1){
    odd=c(odd,A[i])
  }
}
}
for(j in 1:length(B)){
if((B[j]%%2)==0){
  even=c(even,B[j])}
else{
  if((B[j]%%2)==1){
    odd=c(odd,B[j])
  }
}
}
print(unique(sort(even)))
print(unique(sort(odd)))
}
ex12()
```

## EXERCISE 13

### Using the sort ().

```
ex1.1<- function()
{A=c(4,1,8,2,9,6,3)
num=as.numeric(readline(paste("Please, enter a number between",min(A),"and", max(A),": ")))
if(num<min(A) | num>max(A)){
  print("The number is not valid")
}
else
  if(num %in% A){
    print(sort(A))
  }
else
{
  A=c(A,num)
  print(sort(A))
}
}
ex1.1()
```

### Without using the sort ().

```
ex1.2<- function()
{A=c(4,1,8,2,9,6,3)
num=as.numeric(readline(paste("Please, enter a number between",min(A),"and", max(A),": ")))
if(num<min(A) | num>max(A)){
  print("The number is not valid")
}
else if(num %in% A){
  for(i in 1:(length(A)-1)){
```

```
  for(j in (i+1):length(A)){
   if(A[i]>A[j]){
    A[c(i,j)] <- A[c(j,i)]
   }
  }
 }
 print(A)
}
else{
 A=c(A,num)
 if(num %in% A){
  for(i in 1:(length(A)-1)){
   for(j in (i+1):length(A)){
    if(A[i]>A[j]){
     A[c(i,j)] <- A[c(j,i)]
    }
   }
  }
 }
 print(A)
}
}
ex1.2()
```

**EXERCISE 14**

```
Calculatedata<-function()
 {grades<-array(c(8,3,2,9,8,6),c(3,2,1))
dimnames(grades)=list(c("Gema","Rosa","Lucia"),c("Programming","Math"))
print(grades)

for(i in 1:ncol(grades)){
 sub <- colnames(grades)[i]
 print(paste("The max and min grade of",sub, "are", max(grades[,i,]),"and", min(grades[,i,])))
 i = i + 1
}
print(paste("The max and min grade of both subjects are: ", max(grades), "and", min(grades)))

print(length(subset(grades, grades[,1,] >=5 & grades[,2,] >= 5)) / ncol(grades))

ask=as.numeric(readline(prompt="Please, enter a grade: "))
if(ask>=0 & ask<=10){
 print(length(which(grades %in% ask)))
}
}
Calculatedata()
```

**EXERCISE 15**

```
Hit<-function()
{A=c(2,3,6,1,7,1)
v = array("*",length(A))
print(A)
print(v)
while ("*" %in% v){
num=as.numeric(readline(prompt="Please, enter a number between 1 and 10: "))
if(num<1 | num>10){
  print("The number is not valid.")
}
else
  if(num %in% A){
    v[which(A %in% num)] <- as.character(num)
    A[which(A %in% num)] <- "*"
    print(A)
    print(v)
  }
else{
  print(A)
  print(v)
}
}
}
Hit()
```

**EXERCISE 16**

**With while**

```
CalculateHeightandWeights<-function()
{data=array(c(1.70,1.65,1.80,70,70,76),c(3,2,1))
dimnames(data)=list(c("Pepe","John","Antonio"),c("Height","Weight"))
print(data)
 i=1
while(i<ncol(data)){
  print(paste("The minimum height and weight are: ", max(data), "and", min(data)))
  i=i+1
}
h=0
D=apply(data,2,mean)
i=1

while(i<=nrow(data)){

  if(data[i,1,]>D[1])

    h=h+1

    i=i+1

}

print(D)

print(h)

j=1

w=0

D=apply(data,2,mean)

while(j<=nrow(data)){

  if(data[j,2,]>D[2])

    w=w+1

  j=j+1


}

print(w)

}

CalculateHeightandWeights()
```

**<u>With for</u>**

```
CalculateHeightandWeights<-function()

{data=array(c(1.70,1.65,1.80,70,70,76),c(3,2,1))

dimnames(data)=list(c("Pepe","John","Antonio"),c("Height","Weight"))

print(data)

for(i in 1:(ncol(data)-1)){

  print(paste("The minimum height and weight are: ", max(data), "and", min(data)))

}

h=0
D=apply(data,2,mean)
for(i in 1:nrow(data)){
  if(data[i,1,]>D[1])
    h=h+1
}
print(D)
print(h)
j=1
w=0
D=apply(data,2,mean)
for(j in 1:nrow(data)){
  if(data[j,2,]>D[2])
    w=w+1
}
print(w)
}
CalculateHeightandWeights()
```

**With repeat**

```
CalculateHeightandWeights<-function()
{data=array(c(1.70,1.65,1.80,70,70,76),c(3,2,1))
dimnames(data)=list(c("Pepe","John","Antonio"),c("Height","Weight"))
print(data)
i=1
repeat{
  print(paste("The minimum height and weight are: ", max(data), "and", min(data)))
  i=i+1
  if(i>=ncol(data))
    break
}
h=0
D=apply(data,2,mean)
i=1
repeat{
  if(data[i,1,]>D[1])
    h=h+1
  i=i+1
  if(i>nrow(data))
    break
}
print(D)
print(h)
j=1
w=0
D=apply(data,2,mean)
repeat{
  if(data[j,2,]>D[2])
    w=w+1
  j=j+1
  if(j>nrow(data))
    break
 }
print(w)
}
CalculateHeightandWeights()
```